

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA:
INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de:
INGENIERO DE SISTEMAS**

**TEMA:
RECONOCIMIENTO AUTOMÁTICO DE LA PLACA DE UN VEHÍCULO DE
ECUADOR**

**AUTOR:
MILTON DANIEL TORRES CARRERA**

**TUTOR:
HOLGER RAÚL ORTEGA MARTÍNEZ**

Quito, febrero de 2020

CESIÓN DE DERECHOS DE AUTOR

Yo, MILTON DANIEL TORRES CARRERA, con documento de identificación N°1717588683, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de grado intitulado: "RECONOCIMIENTO AUTOMÁTICO DE LA PLACA DE UN VEHÍCULO DE ECUADOR", mismo que ha sido desarrollado para optar por el título de: INGENIERO DE SISTEMAS, en la Universidad Politécnica Salesiana, que dando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservó los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en digital a la Biblioteca de la Universidad Politécnica Salesiana.



Milton Daniel Torres Carrera

CI: 1717588683

Quito, febrero de 2020

DECLARATORÍA DE COAUTORÍA DEL DOCENTE TUTOR

Yo declaro que bajo mi dirección y asesoría fue desarrollado el artículo académico, “RECONOCIMIENTO AUTOMÁTICO DE LA PLACA DE UN VEHÍCULO DE ECUADOR” realizado por MILTON DANIEL TORRES CARRERA, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Quito, febrero de 2020.



Holger Raúl Ortega Martínez.

CI: 1708182728

DEDICATORIA

Dedico este trabajo a Dios que me ha dado las mujeres más importantes de mi vida (Mami, Esposa y Hermana), quienes han hecho esto posible ya que cada una de ellas ha aportado junto a Dios la fortaleza y tenacidad que necesitaba para cumplir esta gran meta, por y para ustedes que siempre han sido mi motivación.

A mi sobrino querido que con sus locuras ha hecho que me esfuerza cada vez más ya que de pronto él podría seguir mis pasos y hay que pisar fuerte.

A mi familia querida quienes han aportado con un granito de arena para ser el hombre que soy.

Y a ti esposa mía que, sin tu esfuerzo, tenacidad y por sobre todo mucha paciencia has encontrado la manera de empujarme de una u otra manera para alcanzar esta meta, muchas gracias.

Milton Daniel Torres Carrera

AGRADECIMIENTOS

Agradezco a la Universidad Politécnica Salesiana que ha contribuido con mi formación profesional, y a mi gran tutor Holger Ortega, que ha logrado una gran motivación para poder realizar este trabajo, y por su profesionalismo, pero más por su excelente calidad de ser humano, gracias profe.

Milton Daniel Torres Carrera

RECONOCIMIENTO AUTOMÁTICO DE LA PLACA DE UN VEHÍCULO DE ECUADOR

AUTOMATIC RECOGNITION OF THE PLATE OF AN ECUADOR VEHICLE GLISH

Holger R. Ortega¹, Milton D. Torres²

Resumen

En el presente proyecto se desarrolla un sistema que tiene como objetivo el reconocimiento de placas vehículos de Ecuador, usando técnicas de visión por computador. Como parte del pre-procesamiento de las imágenes se ha realizado redimensionamiento y binarización sobre las imágenes originales.

Para la segmentación de la imagen se utilizó, detección de bordes, componentes conectados y un algoritmo preliminar para la identificar la ubicación de las letras y los números de la placa. El reconocimiento automático se realiza mediante dos redes neuronales sobre las secciones consideradas como "caracteres", teniendo en cuenta las particularidades de las placas ecuatorianas. Los errores obtenidos bordean el 3%.

Palabras Clave: algoritmo, procesamiento digital de imágenes, visión artificial, redes neuronales, imagen digital, compresión de imágenes.

Abstract

This project, a system is developed that aims to recognize vehicle license plates in Ecuador, using computer vision techniques. As part of the preprocessing of the images, resizing and binarization have been carried out on the original images.

For image segmentation, edge detection, connected components and a preliminary algorithm will be identified to identify the location of the letters and numbers on the board. Automatic recognition is carried out through two neural networks on the sections considered as "characters", taking into account the peculiarities of the Ecuadorian plates. The errors allowed bordering 3%.

Keywords: algorithm, digital image processing, artificial vision, neural networks, digital image, image compression.

¹ Docente de la Carrera de Ingeniería de Sistemas – Universidad Politécnica Salesiana, Sede Quito – Campus Sur.
Autor para correspondencia: hortega@ups.edu.ec

² Estudiante de Ingeniería de Sistemas – Universidad Politécnica Salesiana, Sede Quito – Campus Sur.
Autor para correspondencia: mtorresca@est.ups.edu.ec

1. Introducción

Esta investigación tiene como fin el desarrollo de un sistema alternativo a los sistemas de reconocimiento de placas vehiculares del mercado, que sirva como base para nuevos proyectos o investigaciones que realicen el reconocimiento automático de una placa vehicular del Ecuador.

Considerando el incremento acelerado del parque automotor del país y que la normativa vigente de Ecuador exige que todo vehículo motorizado debe tener una placa que lo identifique para poder circular es importante automatizar el proceso de identificación de las placas vehiculares a partir del reconocimiento digital de la misma y esta manera usarla para muchas aplicaciones tales como control de acceso, infracciones de tránsito, aplicaciones de tráfico, fronteras, tele peajes, control en carretera vehículos que estén siendo buscados por la justicia, entre otros, procesos que realizándolos de manera manual llevarían mucho tiempo y por ser una actividad repetitiva estarían propensos a tener errores.

Este proyecto se enmarca dentro del reconocimiento de patrones, específicamente reconocimiento óptico de caracteres, el cual consiste en la identificación de símbolos o caracteres alfabéticos presentes en una imagen digital.

El proyecto se desarrolla a partir de la captura de la parte frontal o posterior de un vehículo a una distancia de 2 a 3 metros y con una placa legible para someter la imagen a un algoritmo de reconocimiento utilizando visión artificial y redes neuronales.

2. Estado del arte

Para el desarrollo de este proyecto se realiza un estudio del estado del arte, resumido en los siguientes puntos:

2.1. Proyectos Relacionados

- “Desarrollo de un sistema de Reconocimiento de Placas Vehiculares, Universidad del Azuay 2015”: Este sistema tiene como objetivo la identificación de los caracteres de la placa vehicular, mediante el análisis de imágenes digitales, se construyó un prototipo

que automatiza el ingreso al parqueadero en conjunto con un interfaz gráfica web que permite la administración, el sistema tiene una tasa de reconocimiento del 91%. [1]

- “Análisis, Diseño e implementación de un sistema de Control de ingreso de vehículos basado en visión artificial y reconocimiento de placas en el parqueadero de la Universidad Politécnica Salesiana” – Sede Cuenca, Universidad Politécnica Salesiana 2014: Este proyecto elaboró un sistema de reconocimiento de placas vehiculares que permita identificar los vehículos que ingresen al parqueadero de la UPS sede Cuenca de una manera más dinámica, lo que permite llevar un control de los vehículos que ingresan a la institución, este proceso se llevó a cabo mediante una cámara colocada en la puerta, conectada a un sistema de visión artificial, el sistema cuenta con un 94% de acierto en el reconocimiento de placas. [2]
- En Perú se tiene el desarrollo del proyecto: “Sistema de reconocimiento de placas y su influencia en la detección de vehículos robados en la municipalidad San Isidro”, en este estudio se sostiene que al realizar la búsqueda de vehículos robados con la policía por los medios tradicionales no se llega ni al 20% de efectividad además de tener un alto costo operativo, por lo que se propone este proceso de automatización en el que se aplica la inteligencia artificial para reconocer los caracteres alfanuméricos de las placas vehiculares, este sistema inicia con la adquisición de una imagen digital a partir de una cámara IP, para posteriormente procesar las imágenes y definir los caracteres y números de la placa, el proyecto está desarrollado en JAVAanpr, la efectividad de este sistema es del 96% [3].
- En México se tienen estudios similares desde el año 2010, con el Proyecto: “Reconocimientos de placas vehiculares en el Instituto Politécnico Nacional”, este sistema parte desde la adquisición de una imagen digital en formato RGB para posteriormente realizar una binarización, segmentación,

filtrado y reconstrucción de caracteres para finalizar con el proceso de segmentación, aunque el sistema fue desarrollado para reconocer las placas vehiculares de México, puede ser empleado en cualquier otro país o estado, el porcentaje de aciertos de este sistema es del 97,27%. [4].

2.2. Experiencias en empresas / Instituciones Ecuatorianas

Puntualmente se tiene el desarrollo de un proyecto que diseña un prototipo para el acceso vehicular en la Universidad Central del Ecuador, de esta manera reemplazar el operador humano que se encargaba de la vigilancia, esto mediante técnicas de visión artificial valiéndose de insumo los videos de vigilancia de acceso del acceso vehicular de la Av. América de la Sede Quito, este sistema tiene un 84,37% de aciertos de las placas reconocidas por este algoritmo. [5]

2.3. Software y Hardware Comercial de reconocimiento de placas

- Transcore: software diseñado para controlar automáticamente procesos operativos en Terminales, depósitos fiscales donde se identifica mediante tecnología RFID, OCR y LPR el paso de vehículos, camiones y contenedores. [6]
- Intellisoft Parking: Permite el reconocimiento de placas vehiculares, el sistema almacena fotos los vehículos y el contenido de la placa, esta información es almacenada por el sistema para la consulta, análisis y depuración posterior, utiliza cámaras que además de poder trabajar 24 horas sin interrupción pueden hacerlo en oscuridad total gracias a la iluminación infrarroja con la que está equipada. [7]
- FxCAM IBW_2000: este sistema está compuesto por un algoritmo para la identificación de placas vehiculares y una cámara integrada.
- 3LPR: es un sistema de placas vehiculares, identifica y registra los vehículos que acceden o salen de un aparcamiento, puede ser utilizado en todas las instituciones que deseen

controlar, vigilar y tener un registro de todos los vehículos que traspasan un determinado acceso, está compuesto por una caja negra muy portable que se encarga de disparar múltiples fotografías para detectar el vehículo, tiene también un algoritmo incorporado que realiza el reconocimiento automático de las imágenes. [8]

3. Marco Teórico

3.1. Etapas visión artificial

La visión artificial tiene el propósito de interpretar la información que hay en una imagen, a imitación del cerebro de los animales. El proceso consta de varias etapas: [9]

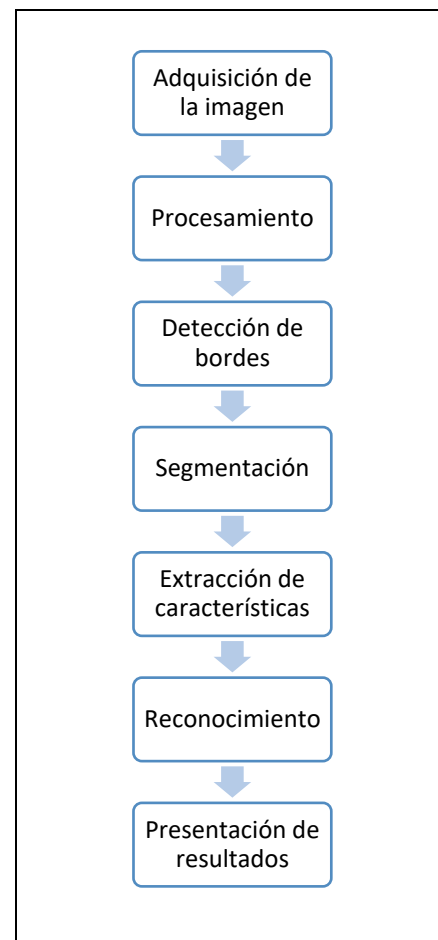


Figura 1. Proceso de visión artificial.

- Adquisición de la imagen: esta etapa consiste en capturar una imagen tridimensional y

convertirla en una imagen digital, mediante un dispositivo de captura. [9]

- **Procesamiento:** en esta fase se ejecutan algoritmos que consisten en filtrar información no deseada o amplificar las características específicas de la imagen y así facilitar los procesos de las siguientes etapas. [9]
- **Detección de bordes.** Inicialmente es necesario encontrar en toda la imagen la región de interés por lo que se desarrolla un algoritmo de detección de contornos, buscando los cambios de tonalidad, mediante este proceso se pueden implementar posteriormente filtros según criterios de clasificación. [9]
- **Segmentación:** en esta etapa se aíslan los elementos que componen la imagen, para determinar las regiones de pixeles necesarios en una escena y de esta forma comprender mejor la imagen y las regiones de interés. [9]
- **Extracción de características:** en esta etapa consiste en utilizar los elementos segmentados de la imagen principal y representar matemáticamente esta información. [9]
- **Reconocimiento:** se clasifican los objetos que aparecen en las imágenes segmentadas [9]
- **Presentación de resultados:** es el objetivo final del proceso de visión artificial, donde se determina que es lo que contiene la imagen general y se lo presenta en una interfaz. [9]

3.2. Binarización

La binarización consiste en transformar una imagen en escala de grises a una imagen blanco y negro, mediante la aplicación de un umbral. [10]

En su gran mayoría las técnicas de binarización se basan en estadísticas sobre el histograma de una imagen para calcular el umbral más adecuado o se lo coloca de manera manual si el entorno donde se captura la imagen está controlado.

Uno de los métodos más utilizados para hallar el umbral correcto de una manera autónoma es el método de Otsu, el cual se lo calcula de la varianza de la luminosidad de los pixeles de la imagen y con esto este valor. [10]

La expresión matemática de este método es:

$$\mu_r = \sum_{q=0}^{L-1} q * \frac{P_q(r_q)}{w_T}$$

Ecuación 1. Método de OTSU

3.3. Componentes Conectados

Los componentes conectados tienen como finalidad agrupar pixeles de las imágenes que componen un objeto, en este caso puntual letras o números. [11]

Para identificar la vecindad o conectividad de un pixel p ubicado en las coordenadas (x, y) , posee 4 vecindades:

- 2 horizontales.
- 2 verticales.

Cuyas coordenadas están descritas por la siguiente ecuación:

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)$$

Ecuación 2. Vecindad de 4.

La vecindad o conectividad diagonal, se describe con el pixel p en (x, y) y su ecuación es:

$$(x + 1, y + 1), (x + 1, y - 1), (x - 1, y + 1), (x - 1, y - 1)$$

Ecuación 3. Vecindad Diagonal

Es importante tener en cuenta la adyacencia el cual existe si dos pixeles tienen en común una de sus fronteras, o al menos una de sus esquinas. [11].

La figura muestra pixeles adyacentes.

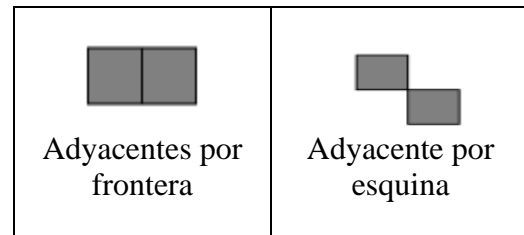


Figura 2. Pixeles adyacentes

Una vecindad de un pixel p_o , denotada como V_p , es una submatriz M_{KL} de tamaño $K \times L$, con K y L enteros impares pequeños, contenida en la matriz

imagen (i_{MN}), la cual está formada por un número finito de pixeles vecinos o no de p_o .

$$V_p = \{p: p \in M_{KL}\}; M_{KL} \subset i_{MN}; K = L = 3,5,9$$

Ecuación 4. Vecindad de un pixel

Las definiciones antes mencionadas son más claras con la siguiente figura:

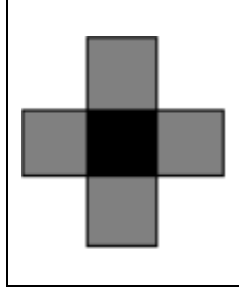


Figura 3. Vecindades de 4

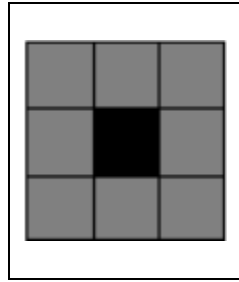


Figura 4. Vecindades de 8

3.4. Redes Neuronales Artificiales (RNA)

Las redes neuronales artificiales o RNAs, son ecuaciones matemáticas que imitan el comportamiento del cerebro biológico, que se caracteriza por aprender en función a la experiencia o la interpretación de un conjunto de datos, la arquitectura de una RNA, tiene los siguientes elementos: capa de entrada, capas ocultas, capa de salida. [13]

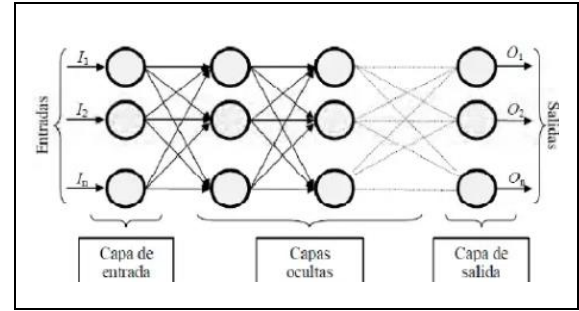


Figura 5. Arquitectura de una RNA

Capa de entrada, señales de entrada, pueden ser discretas o continuas.

Pesos Sinápticos, representan la intensidad de interacción, estos pesos pueden ser variables o fijos.

Regla de propagación, a partir de las entradas y los pesos permite obtener como resultado el valor del potencial entre las neuronas.

Función de activación, toma también el nombre de función de transferencia, proporciona el estado actual de la activación en función a su anterior estado y de la anterior neurona, para posteriormente entregar la salida global de la neurona [14], en este trabajo se utiliza la función sigmoide.

La función sigmoide transforma los valores ingresados en una escala de 0 a 1, donde los valores altos tienen a valorarse en 1 y los muy bajos a 0.

Características de la función sigmoide:

- EL gradiente se satura y elimina.
- La convergencia es lenta.
- No está centrada en el cero.
- Está acotada entre 0 y 1.
- La última capa presenta un alto rendimiento.

$$f(x) = \frac{1}{1 + e^{-x}}$$

Ecuación 5. Función sigmoide

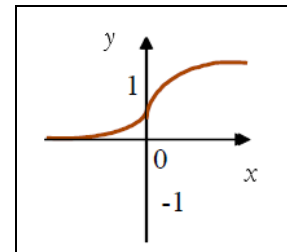


Figura 6. Función de activación sigmoide

Al entrenar la red neuronal artificial, además del error obtenido respecto al conjunto de patrones de aprendizaje, debe considerarse el error de generalización, el cual debe medirse con una base de muestras diferente al que se utilizó para el proceso de entrenamiento.

Por lo que toma mayor importancia una buena generalización de la red que conseguir un error muy pequeño en la submuestra de entrenamiento, esto indicara que el sistema ha desarrollado de forma correcta las relaciones subyacentes en los datos. [13]

Backpropagation: es un método de cálculo del gradiente usado para realizar el entrenamiento de redes neuronales artificiales. Se aplica un estímulo a partir de un patrón de entrada para que se propague desde la primera capa a través de las capas internas de la red, generando finalmente una salida.

Para el cálculo de la señal de error se lo realiza comparando el valor de la señal de entrada con el valor de la señal de salida. [15]

A partir de la capa de salida, el error se propaga hacia atrás, es decir se distribuye entre las capas ocultas que afectaron la salida, pero la neurona de la capa oculta, adquiere una parte proporcional de la señal del error, tomando en cuenta la contribución relativa de aportación de cada neurona a la salida inicial.

Este proceso se repite, capa por capa, de esta manera las neuronas que componen la red neuronal artificial, se les asigna un valor de error con respecto al valor del error global. [15]

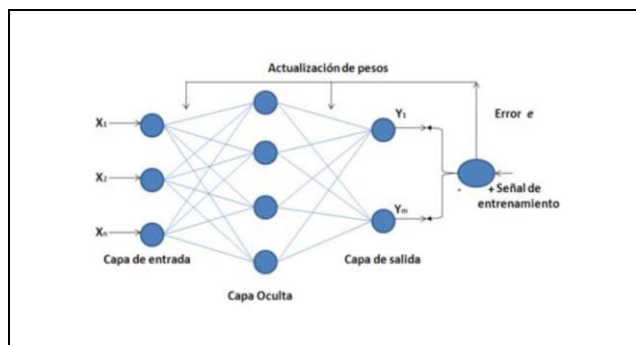


Figura 7. Red neuronal: backpropagation

3.5. C#

“C#, es un lenguaje de desarrollo de programación construido por Microsoft. Es un lenguaje basado en la sintaxis de C y C++. Este lenguaje de

programación C#, debido a que es un lenguaje independiente existen compiladores que generan compilaciones para distintas plataformas, además está orientado a objetos.” [16]

3.6. EMGU CV

“Es un framework para .Net, basado en la librería de OPENCV, el cual está desarrollado en C# distribución que permite utilizar las funciones de la librería de procesamiento de imágenes OpenCV, para poder ser usadas con otros lenguajes de programación tales como: C#, VB, VC++, Iron Python. El framework puede ser compilada en MonoDeveloper y ejecutada en Windows, Linux, Mac OS X y dispositivos Android.” [17]

3.7. Matlab

“MATLAB es un IDE, que combina un entorno de escritorio y los procesos de diseño con un lenguaje de programación de alto nivel propio (Lenguaje M), que permite el procesamiento matemático de matrices y arrays directamente. El entorno de MATLAB posee APP's, que permiten visualizar el funcionamiento de algoritmos con sus datos.” [18]

4. Materiales y Métodos

4.1. Herramientas

Para este proyecto se utiliza el lenguaje de programación C#, donde se desarrollan distintos algoritmos para el procesamiento digital de imágenes el cual consta de 2 etapas, la primera consiste en el proceso de extracción de números y letras de la placa del vehículo y la segunda, consta en la evaluación de la placa utilizando las redes neuronales desarrolladas en Matlab y el clasificador para mostrar los resultados de las redes neuronales en caracteres alfanuméricos.

El diagrama de funcionamiento se describe en el siguiente gráfico:

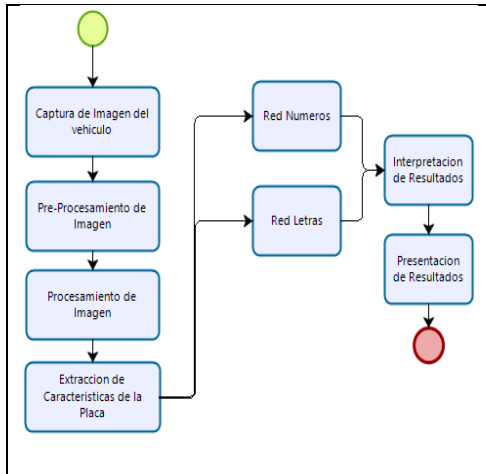


Figura 8. Diagrama de funcionamiento del sistema

4.2. Captura de imágenes

Parte fundamental de este proyecto es la captura de la imagen del vehículo ya sea de la parte frontal o posterior donde se encuentre la placa del vehículo, este proceso se lo realiza utilizando la cámara de un dispositivo móvil de marca Samsung modelo S8 con un lente óptico de 16 megapíxeles y una resolución de 4032 x 3024.

Las imágenes se toman a una distancia de 2 a 3 metros, tanto de la parte anterior como de la posterior del vehículo.

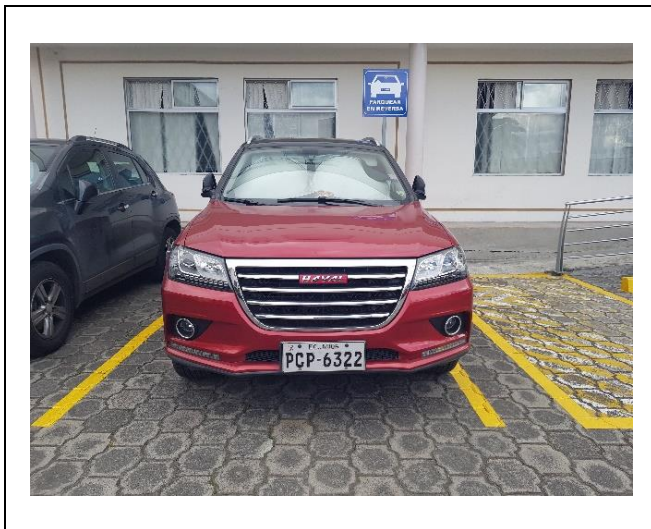


Figura 9. Fotografía de un vehículo de la parte frontal. (Imagen HD 4032 x 3024 píxeles)



Figura 10. Fotografía de un vehículo de la parte posterior. (Imagen HD 4032 x 3024 píxeles)

4.3. Preprocesamiento y Procesamiento

Las imágenes, originalmente de 4032 x 3024 píxeles, fueron reducidas a una resolución de 800 x 600 píxeles. Empíricamente se comprobó que esta es una resolución suficiente para realizar la posterior clasificación: partiendo de la resolución original, se fue bajando esta gradualmente entre las resoluciones más comunes (1920x1080, 1366x768, 1280x800, 800x600) donde se elige la resolución 800x600 con la que se tiene un bajo costo computacional se reduce.

Para valores menores de resolución la clasificación resultó ineficiente.



Figura 11. Imagen reducida la resolución. (Imagen 800 x 600 píxeles)



Figura 12. Imagen reducida la resolución. (Imagen 800 x 600 pixeles)

Sobre estas imágenes se realiza una transformación a escala de grises, pues el color no resulta una característica discriminante para el reconocimiento de caracteres.



Figura 13. Imagen viste frontal en escala de gris



Figura 14. Imagen vista posterior en escala de gris

En las Figuras 11 y 11 se observa el resultado de la reducción de resolución y en las Figuras 13 y 14 el resultado de la transformación a escala de grises.

4.4. Binarización OTSU

Una vez realizada la reducción de resolución y la transformación a escala de grises, se procede con la binarización OTSU, con el objetivo de reducir la cantidad de información a procesar y agilizar el proceso de segmentación. Empíricamente se establece que esta reducción no es en detrimento de la capacidad del sistema para reconocer los caracteres en la imagen.



Figura 15. Binarización OTSU, vista frontal



Figura 16. Binarización OTSU, vista posterior

En las Figuras 15 y 16 se muestra el proceso de binarización OTSU, donde la placa resalta en la imagen y esto facilita los siguientes procesos.

4.5. Componentes conectados

Para facilitar la segmentación de la parte de la imagen correspondiente a la placa se usa el método “FindContours”, de la librería “Emgu.CV”, que implementa el algoritmo de componentes conectados.



Figura 17. Componentes conectados, vista frontal



Figura 18. Componentes conectados, vista posterior

En las Figuras 17 y 18 se aprecia el resultado de aplicar componentes conectados a la imagen: se observan los bordes destacados de objetos como faros, parabrisas, y la misma placa del auto.

4.6. Segmentación

La segmentación se realiza aplicando criterios discriminatorios como la ubicación de la placa, que por lo general se encuentra en la mitad inferior del vehículo. Por otra parte, la distancia estándar de 2 a 3 metros a la que se obtuvieron las fotos, los caracteres de las letras y los números mantienen una cantidad de píxeles normada tanto en su alto y ancho.

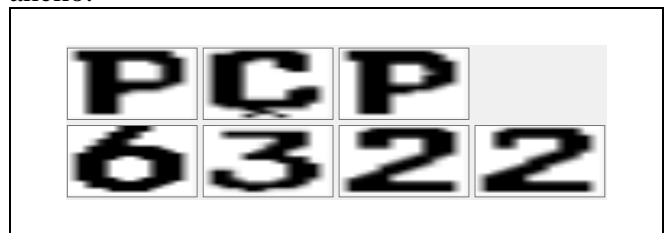


Figura 19. Segmentación de letras y números

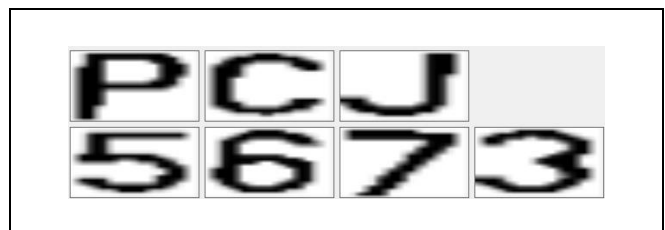


Figura 20. Segmentación de letras y números

Aplicando estos criterios se obtiene una segmentación eficiente, ejemplificada en las figuras 19 y 20. Donde se puede identificar que cada segmento extraído corresponde a una letra o número, representado la placa vehicular.

4.7. Clasificación

Para la clasificación se usan dos redes neuronales, las cuales se desarrollan en Matlab, específicamente se utiliza la herramienta “Neural Net Pattern Recognition” que permite realizar el entrenamiento de las redes neuronales con las muestras obtenidas en la etapa de segmentación.

Una red neuronal se utiliza para clasificar letras y otra para números. Se toma en cuenta la particularidad de las placas de Ecuador donde los primeros tres caracteres deben ser letras y los siguientes números. Las redes neuronales utilizadas tienen una capa oculta y la función de activación sigmoide tanto en las capas internas como en la de salida.

La implementación corresponde a las funciones “RLetras” y “RNumeros” de las librerías desarrolladas en Matlab “Red Numeros” y “Red Letras”.

Para el entrenamiento de las redes neuronales se usa un conjunto de 500 fotos de letras y 500 de números, es decir un aproximado de 50 ejemplos por cada carácter numérico y 20 por cada letra.

Previamente, en el proceso de extracción de características las imágenes segmentadas, son reducidas a la resolución 20x20 pixeles, y transformadas a un vector de características de dimensión 400, con el que se alimenta la red junto con las correspondientes etiquetas de clase.

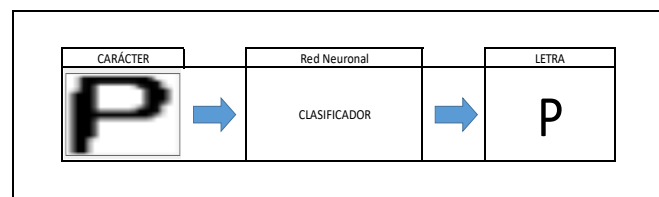


Figura 21. Clasificador letras

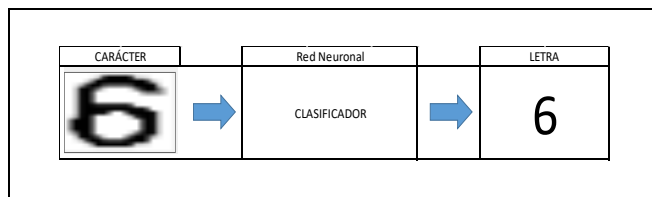


Figura 22. Clasificador números

En las figuras 21 y 22 se aprecia el trabajo del clasificador de letras y números.

Consolidando la información arrojada por ambos clasificadores, el sistema es capaz de devolver la información completa sobre la placa del vehículo, tal como se muestra en las figuras 23 y 24.



Figura 23. Resultado Final ejemplo 1



Figura 24. Resultado Final, ejemplo 2

5. Resultados y Discusión

Para evaluar la eficiencia de la clasificación se utilizan aproximadamente 150 fotografías tomadas a una distancia aproximada de 2 a 3 metros del vehículo con una cámara digital de celular marca Samsung, modelo S8, de 12 megapíxeles; es decir, la misma que se utilizó para generar el conjunto de entrenamiento.

Tabla 1. Muestra de números

Núm.	# Muestras	Núm.	# Muestras	Núm.	# Muestras
0	52	4	56	8	46
1	57	5	57	9	64
2	61	6	55		
3	59	7	46		
Total				553	

Tabla 2. Muestra de letras

Letras	# Muestras	Letras	# Muestras	Letras	# Muestras
A	16	J	16	S	16
B	13	K	16	T	16
C	15	L	15	U	14
D	16	M	13	V	14
E	14	N	11	W	8
F	13	O	13	X	10
G	16	P	78	Y	12
H	14	Q	14	Z	10
I	14	R	15		
Total				131	

La matriz de confusión obtenida para el clasificador de números es la siguiente:

Tabla 3. Matriz de confusión de números

Núm.	0	1	2	3	4	5	6	7	8	9
0	52	0	0	0	0	0	0	0	0	64
1	0	57	0	0	0	0	0	0	0	58
2	0	0	61	0	0	0	0	0	0	52
3	0	0	0	59	0	0	0	0	0	46
4	0	0	0	0	56	0	0	0	0	40
5	0	0	0	0	0	57	0	0	0	24
6	0	0	0	0	0	0	55	0	0	18
7	0	0	0	0	0	0	0	46	0	12
8	0	0	0	0	0	0	0	0	46	6
9	0	0	0	0	0	0	0	0	0	64

Para el clasificador de letras, la matriz de clasificación obtenida es:

Matriz de confusión Letras																										
Letr.	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78
B	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	72
C	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	69
D	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	66
E	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	63
F	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60
G	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	57
H	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	54
I	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	51
J	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	48
K	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45
L	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	42
M	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	39
N	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	2	0	0	0	36
O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	33
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78	0	0	0	0	0	0	0	0	0	30
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	27
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0	24
S	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	21
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	18
U	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	15
V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	12
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	9
X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	6
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	3
Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10








Figura 25.Matriz de confusión letras

Una muestra de la información ya consolidada que arroja el sistema se puede visualizar a continuación:

Tabla 4. Caracteres no reconocidos

PLACAS NO IDENTIFICADAS		
PLACA	FOTOGRAFIA	N*
PCW9455		W
PCN8211		N

Tabla 5. Muestra de placas identificadas

PLACA	FOTOGRAFIA
PAC4776	
PBX8613	
PBK5309	
PBQ8299	
HBC1882	
GSL8647	
GJF0504	

Por lo que de manera general se obtiene una efectividad del 97% de las 150 placas de prueba, el 3% restante corresponde a las no reconocidas donde falla el reconocimiento el algún carácter, esto se debe a la calidad de la placa y la distancia con la que fue tomada la fotografía.

6. Conclusiones

El resultado de este proyecto es un sistema de reconocimiento automático de placas vehiculares tomando en cuenta las características de una placa estándar del Ecuador. El sistema usa binarización

de Otsu para el procesamiento de las imágenes, el algoritmo de Componentes Conectados para la segmentación y dos redes neuronales para la clasificación (una para la clasificación de letras y otra para la clasificación de números). Las redes neuronales se configuraron y entrenaron usando la *app* correspondiente de MATLAB; el código en general fue escrito en C#.

Las fotos se tomaron en condiciones relativamente controladas: a una distancia estándar de 2 a 3 metros a partir del vehículo; durante el día, para tener iluminación de luz natural; y direccionando la cámara de manera de formar un ángulo perpendicular entre la visual y la superficie de la placa.

Bajo estas condiciones, la tasa de efectividad del sistema es del 97%, lo que se considera normalmente aceptable para una tarea de clasificación resuelta con Machine Learning. El sistema, entonces, podría ser usado como base en una aplicación práctica real, como un sistema de monitoreo y vigilancia a la entrada de un parqueadero institucional, por ejemplo.

Para ello, el modelo debería ser mejorado para responder de manera robusta a cambios como: tipos especiales de placa (cuerpo diplomático, consular, motos, etc.), imágenes tomadas a ángulos oblicuos con respecto a la superficie de la placa, iluminación variable (autos que ingresan a diferentes horas del día o en la noche), etcétera. Además, el sistema desarrollado no incluye la automatización del proceso de captura de imagen, que en cambio sería imprescindible en el sistema definitivo.

Referencias

- [1] D. D. Espinoza Saquicela y C. A. Salinas Escobar, *Desarrollo de un Sistema de Reconocimiento de Placas Vehiculares*, Cuenca, 2015.
- [2] M. A. Álvarez Durán, *Análisis, Diseño e implementación de un sistema de Control de ingreso de vehículos basado en visión artificial y reconocimiento de placas en el parqueadero de la Universidad Politécnica Salesiana – Sede Cuenca*, Cuenca, 2014.

- [3] D. R. Pastrano, «Desarrollo de un Sistema de reconocimiento de placas y su influencia en la detección de vehículos robados en la Municipalidad de San Isidro,» Lima, 2017.
- [4] J. L. D. Montiel, «Reconocimientos de placas vehiculares,» México, 2010.
- [5] G. M. F. Paucar, «Repositorio digital de tesis UCE,» 07 2017. [En línea]. Available: <http://www.dspace.uce.edu.ec/bitstream/25000/11755/1/T-UCE-0011-323.pdf>.
- [6] S. TransCore, «Sic TransCore,» 2019. [En línea]. Available: https://www.sictranscore.com/sicpuertos-sistema-de-identificacion-en-puertos/?gclid=CjwKCAiAluLvBRASEiwAAbX3GSmcBbb70HZls5Qn1Az6sF9jAgeVIBS1lDYwY_YeFC9VUWp_cSwTARoC3xkQAvD_BwE.
- [7] I. Parking, «Intellisoft Parking,» 2013. [En línea]. Available: http://www.intellisoftparking.com/cms/index.php/productos/lpr?gclid=CjwKCAiAluLvBRASEiwAAbX3GYc8JmKqfNWzd iKIRy-l26-aQArldjXd8nvIF0ROOyYby3dStZ7rORoC7KwQAvD_BwE.
- [8] 3. LPR, «Sistema de reconocimiento de patentes vehiculares,» España, 2019.
- [9] J. Velez, *Visión por computador*, Madrid: S.L Dykinson, 2003.
- [10] U. N. d. Quilmes, «Visión Artificial,» Octubre 2015. [En línea]. Available: <https://es.slideshare.net/JorgeAntonioParr aSerquen/segmentacin-por-umbralizacin-mtodo-de-otsu>.
- [11] G. Pasariello, *Imágenes médicas: adquisición, análisis, procesamiento e interpretación*, Venezuela: Equinoccio, 2013.
- [12] A. Carcedo y Franco, *Programa de segmentación de regiones en imágenes médicas en MATLAB*, Puebla, 2004.
- [13] R. Flórez, *Las redes neuronales artificiales*, Lorena Bello, 2008.
- [14] J. L. O. Carrion, *Reconocimiento de caracteres de una placa de automovil mediante redes neuronales artificiales utilizando matlab*, 2011.
- [15] S. Haykin, *Neural Network and Learning Machine*, New Jersey: Editorial Prentice Hall, 2009.
- [16] Microsoft, «Microsoft,» 2020. [En línea]. Available: <https://docs.microsoft.com/en-us/dotnet/csharp/>.
- [17] EMGU, «EMGU,» 2019. [En línea]. Available: http://www.emgu.com/wiki/index.php/Main_Page.
- [18] I. The MathWorks, «Matlab,» 2020. [En línea]. Available: <https://la.mathworks.com/products/matlab.html>.
- [19] J. E. Pérez Villalba, *Reconocimiento de placas vehiculares mediante procesamiento de imagenes para optimizar el acceso a los parqueaderos de la UTA, Campus Huachi*, Ambato, 2014.
- [20] F. S. Caparrini, «Enntrenamiento de redes neuronales,» 2017. [En línea]. Available: <http://www.cs.us.es/~fsancho/?e=165>.